WEB PERFORMANCE

오주의 마법사

김민준 류성탁 이솔잎

메인 페이지 왜 이렇게 느려?

메인 페이지가 로딩되는게 눈에 보여!

이거 좀 버벅거리는 것 같은데요

그래도..메인 페이지인데 좀 더 빠르게 할 수 없을까?

Reservation System Performance

LIGHTHOUSE

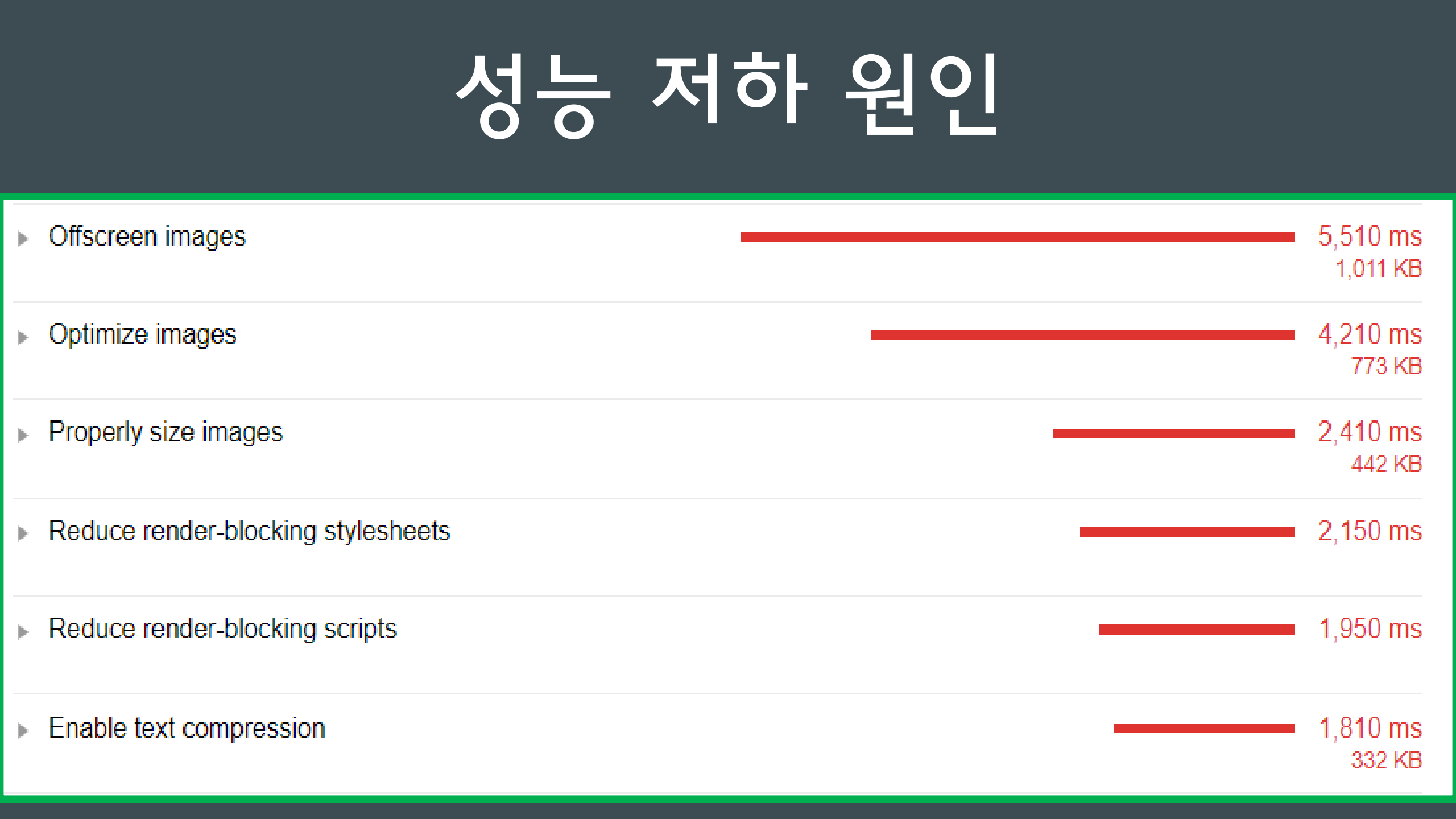
LIGHTHOUSE 실행 환경

Runtime environment

- User agent: **Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/60.0.3112.101 Safari/537.36**
- Device Emulation Nexus 5X: **Enabled**
- Network Throttling 562.5ms RTT, 1.4Mbps down, 0.7Mbps up: **Enabled**
- CPU Throttling 4x slowdown: **Enabled**

모바일 환경
네트워크 562.5ms RTT / 다운로드 속도 1.4Mb / 업로드 속도 0.7Mb
CPU 4배 성능 저하상태

성능 저하 원인


Offscreen images
5,510 ms
1,011 KB
Optimize images
4,210 ms
773 KB
Properly size images
2,410 ms
442 KB
Reduce render-blocking stylesheets
2,150 ms
Reduce render-blocking scripts
1,950 ms
Enable text compression
1,810 ms
332 KB

왜 이것들이 문제가 될까?

불필요하게 **리소스 크기**가 커지면

다운로드 시간이 길어지고

결과적으로 **렌더링 시간**에 영향을 미친다.

첫번째 개선

문제1. Offscreen Images

문제2. Optimize Images

문제3. Properly size Images

문제4. Text Compression

Offscreen Image 란?

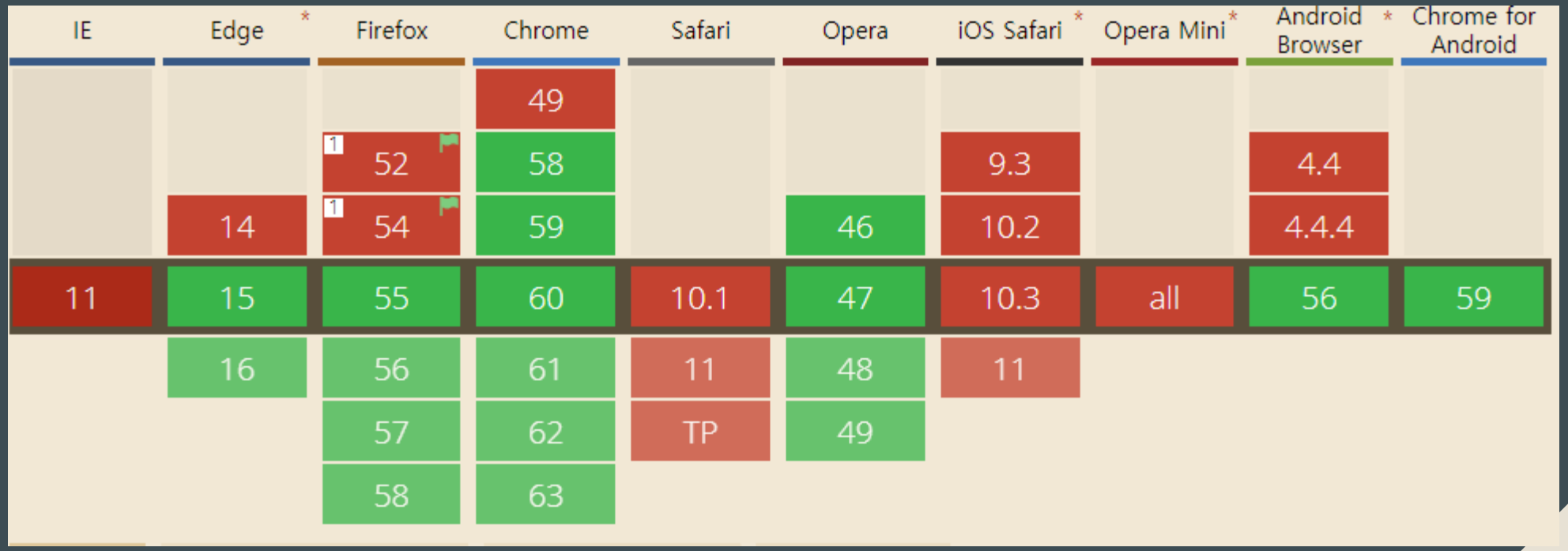
화면에 보이지 않는 이미지

페이지가 로드 되는 시점에서 화면에 보이지 않는 이미지들로 인해서 사용자 입장에서 불필요한 다운로드가 발생된다.

IntersectionObserver

- Observer에 등록된 DOM element들이 화면에 나타나거나 사라질 때 비동기로 이벤트를 발생시킨다.

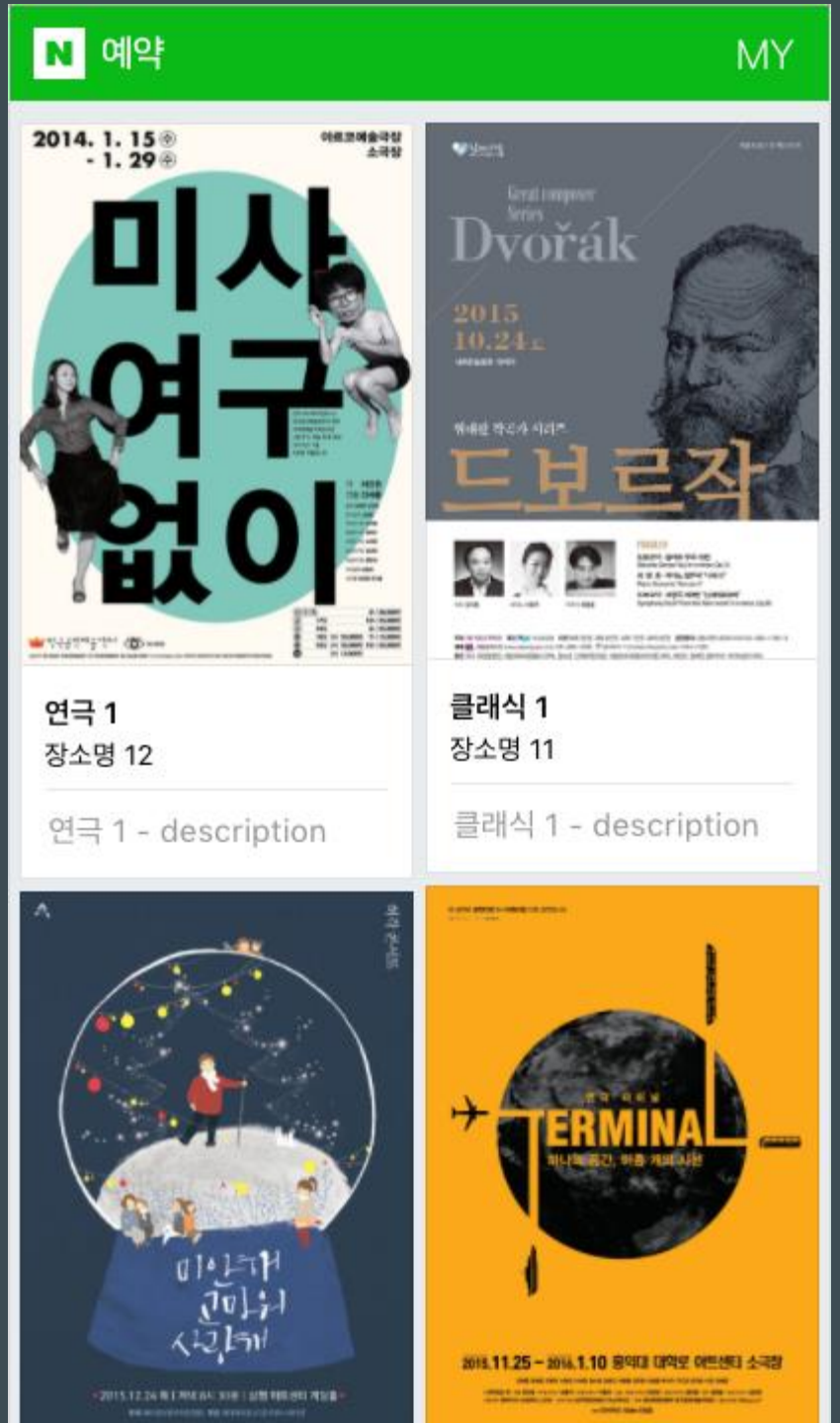
- 브라우저 지원 현황



BEFORE

화면에 보여지는 이미지 뿐만 아니라
화면에 보이지 않는 이미지까지 다운로드

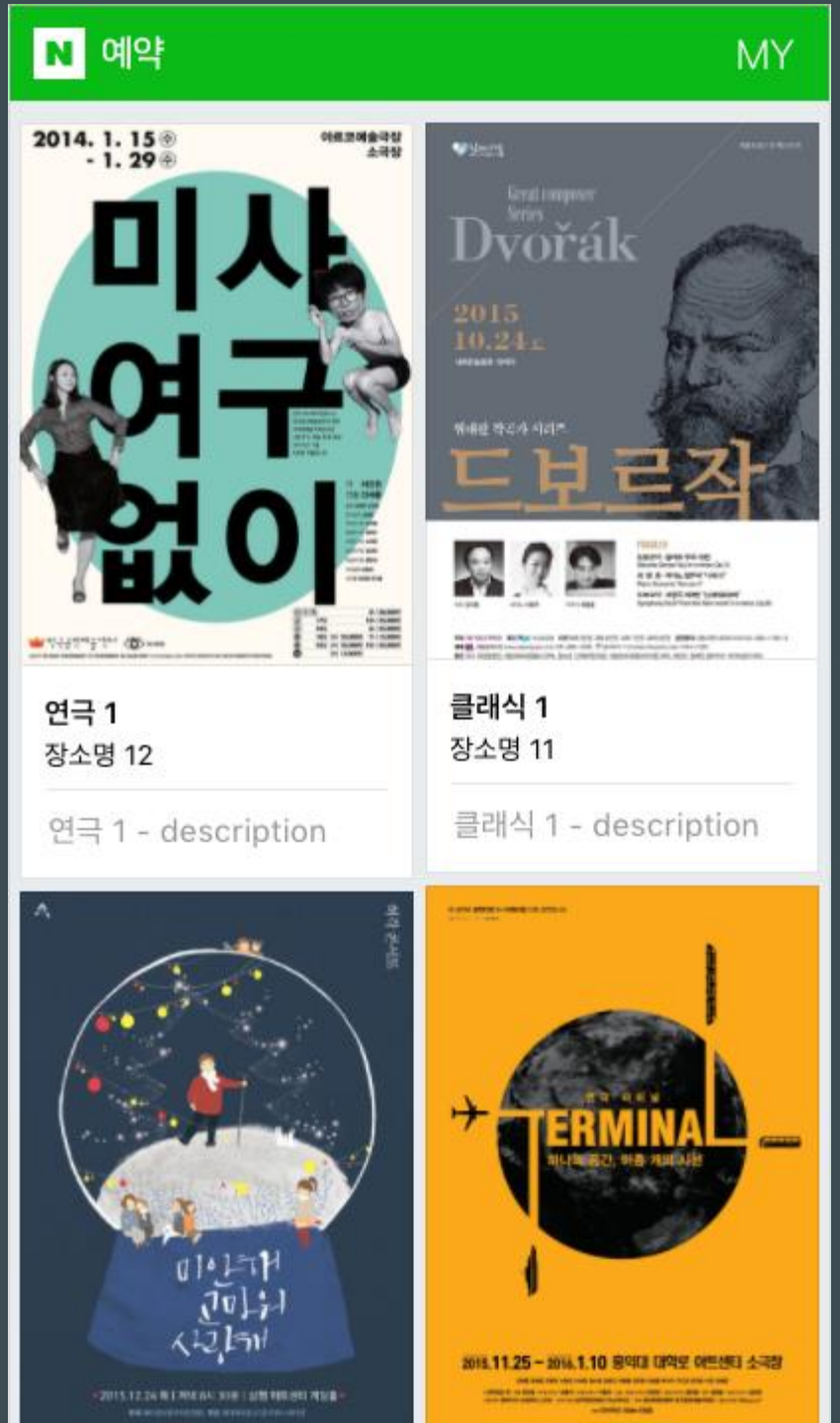
Name	Status	Type	Initiator	Size	Time
spr_book_event.png	200	png	main.bundle.js:14	1.7 KB	11 ms
spr_book2.png	200	png	main.bundle.js:14	55.4 KB	13 ms
spr_bi.png	200	png	main.bundle.js:14	9.3 KB	12 ms
image.jpg	200	jpeg	main.bundle.js:14	277 KB	34 ms
image.jpg	200	jpeg	main.bundle.js:14	176 KB	29 ms
9	200	jpeg	main.bundle.js:24	344 KB	99 ms
18	200	jpeg	main.bundle.js:24	171 KB	178 ms
17	200	jpeg	main.bundle.js:24	255 KB	112 ms
16	200	jpeg	main.bundle.js:24	59.6 KB	16 ms
15	200	jpeg	main.bundle.js:24	219 KB	97 ms
14	200	jpeg	main.bundle.js:24	65.9 KB	15 ms
13	200	jpeg	main.bundle.js:24	220 KB	97 ms
12	200	jpeg	main.bundle.js:24	384 KB	25 ms



AFTER

화면에 보여지는 이미지만 다운로드

Name	Status	Type	Initiator	Size	Time
spr_book_event.png	200	png	(index)	1.7 KB	50 ms
spr_book2.png	200	png	(index)	55.4 KB	51 ms
spr_bi.png	200	png	(index)	9.3 KB	46 ms
image.jpg	200	jpeg	(index)	176 KB	50 ms
image.jpg	200	jpeg	(index)	277 KB	52 ms
18	200	jpeg	main.bundle.js:24	171 KB	38 ms
17	200	jpeg	main.bundle.js:24	255 KB	50 ms
16	200	jpeg	main.bundle.js:24	59.6 KB	27 ms
15	200	jpeg	main.bundle.js:24	219 KB	48 ms



두번째 개선

~~문제1. Offscreen Images~~

문제2. Optimize Images

문제3. Properly size Images

문제4. Text Compression

이미지는 웹페이지에서 다운로드 되는 바이트의 대부분을 차지한다.

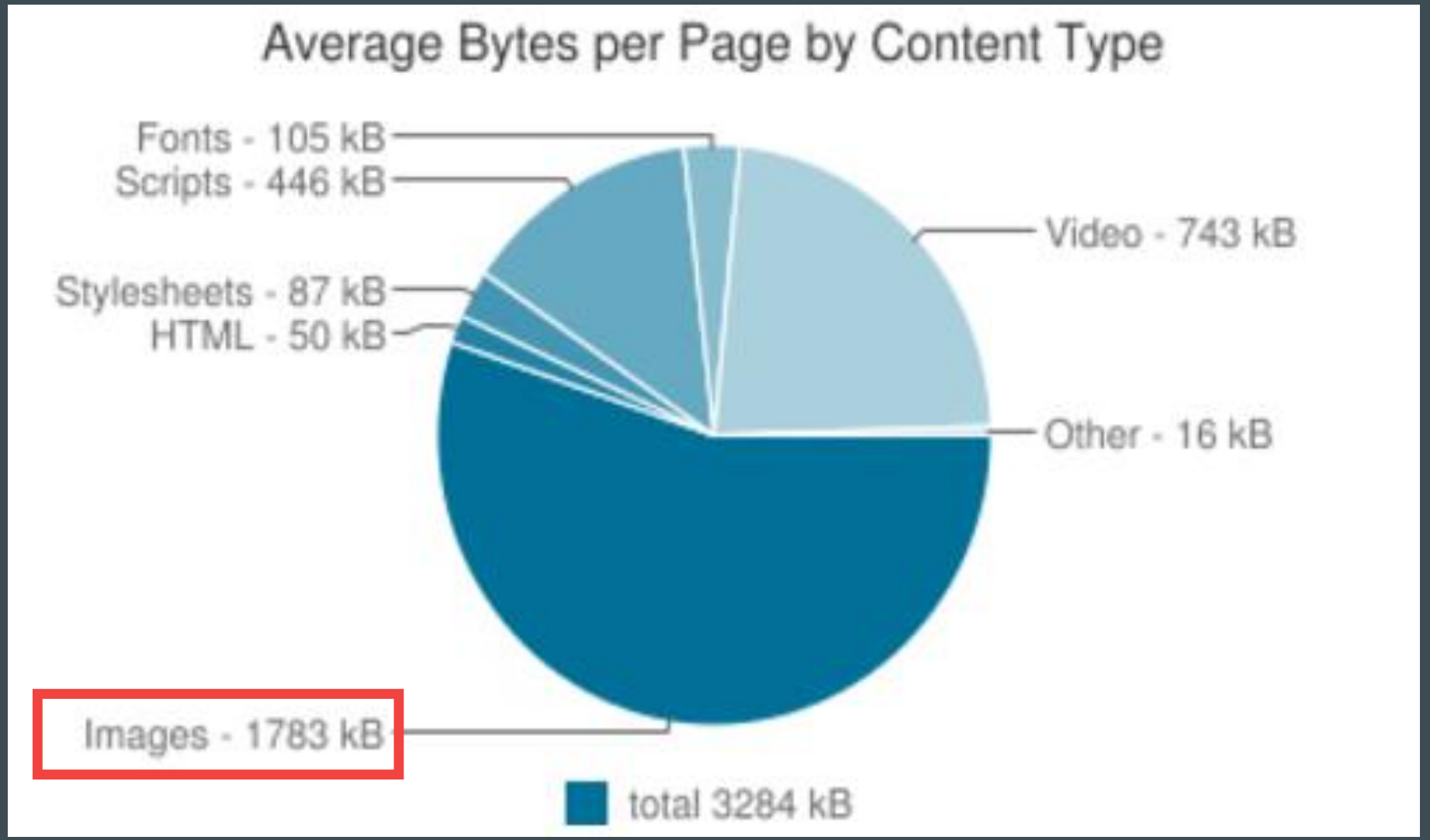


Image Compression

Name	Status	Type	Initiator	Size	Time
spr_book_event.png	200	png	(index)	1.7 KB	50 ms
spr_book2.png	200	png	(index)	55.4 KB	51 ms
spr_bi.png	200	png	(index)	9.3 KB	46 ms
image.jpg	200	jpeg	(index)	176 KB	50 ms
image.jpg	200	jpeg	(index)	277 KB	52 ms
18	200	jpeg	main.bundle.js:24	171 KB	38 ms
17	200	jpeg	main.bundle.js:24	255 KB	50 ms
16	200	jpeg	main.bundle.js:24	59.6 KB	27 ms
15	200	jpeg	main.bundle.js:24	219 KB	48 ms

BEFORE

이미지 크기 : 1,157KB

시간 : 265ms

Name	Status	Type	Initiator	Size	Time
spr_book_event.png	200	png	(index)	1.7 KB	42 ms
spr_book2.png	200	png	(index)	55.4 KB	42 ms
spr_bi.png	200	png	(index)	9.3 KB	25 ms
46	200	jpeg	(index)	58.8 KB	43 ms
45	200	jpeg	(index)	48.1 KB	43 ms
18	200	jpeg	main.bundle.js:24	18.6 KB	23 ms
17	200	jpeg	main.bundle.js:24	23.9 KB	20 ms
16	200	jpeg	main.bundle.js:24	28.4 KB	22 ms
15	200	jpeg	main.bundle.js:24	15.7 KB	15 ms

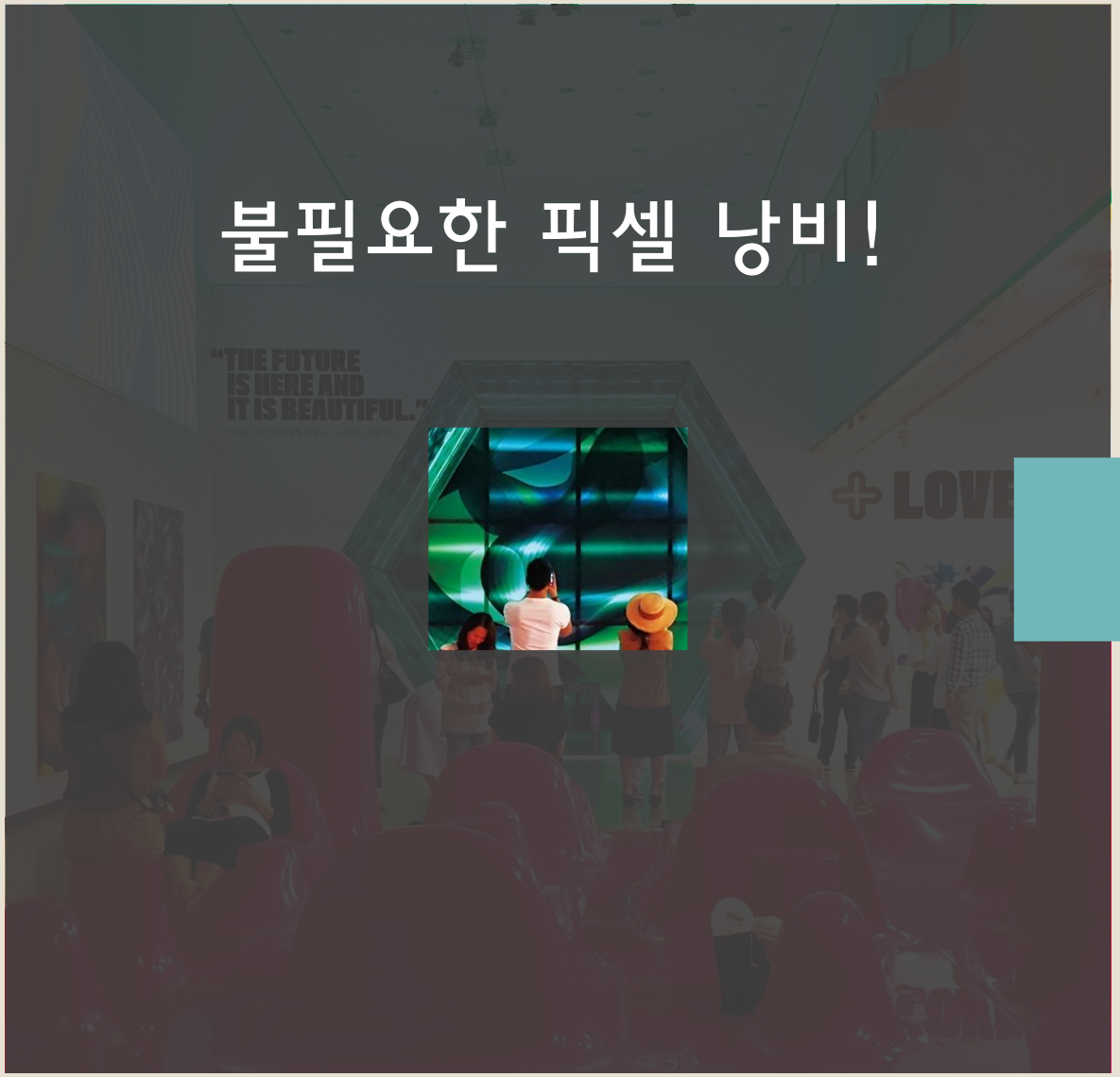
AFTER

이미지 크기 : 190KB

시간 : 166ms

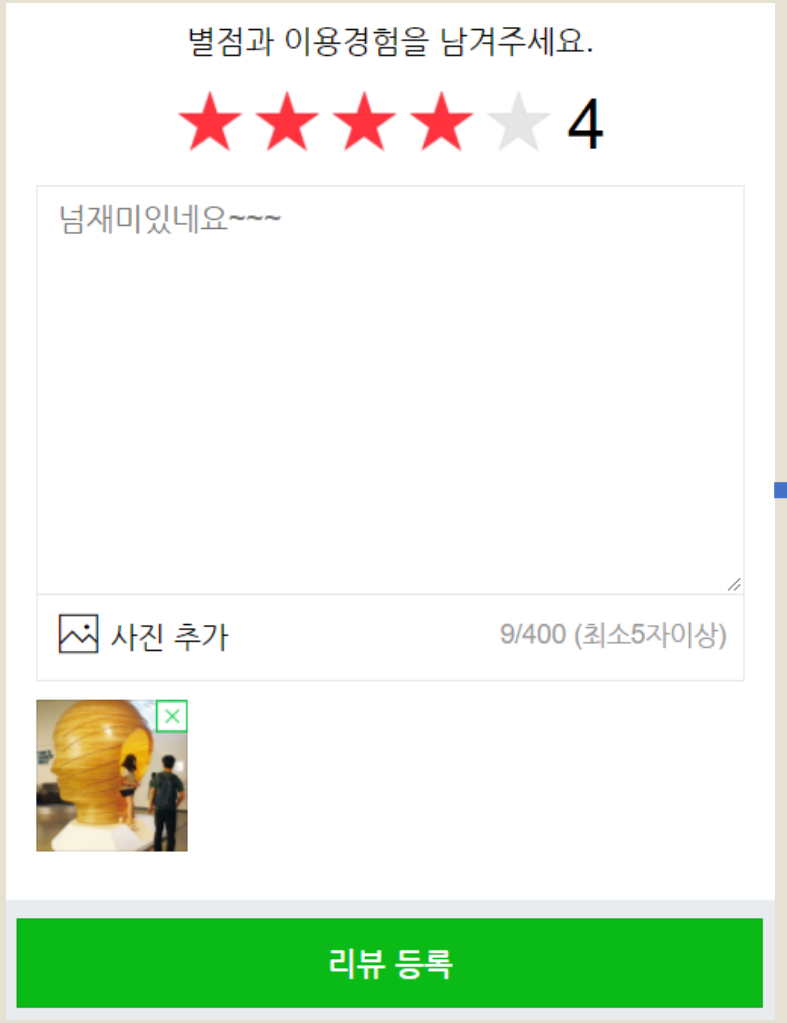
Image Resize

실제 이미지 크기 1500 X 1500



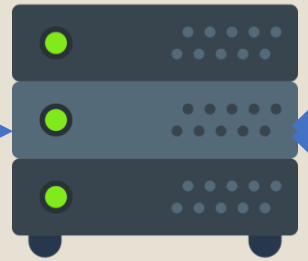
오버헤드 발생!

예매자 한줄평 등록



SERVER

1. ImageIO
2. Thumbnails



원본

압축 이미지

썸네일

1500 x 1500 , 428KB

90 x 90, 3KB

2,247,200 픽셀 감소, 바이트 크기 약 **425KB** 감소

세번째 개선

~~문제1. Offscreen Images~~

~~문제2. Optimize Images~~

~~문제3. Properly size Images~~

문제4. Text Compression

왜 Text Compression을 해야 할까?

개발 시 필요한 주석, 공백, 긴 변수명!

클라이언트에겐 불필요한 리소스!

webpack

style.css	200	stylesheet	(index)	88.1 KB	12 ms
require.js	200	script	(index)	84.7 KB	30 ms
spr_bi.png	200	png	(index)	9.3 KB	48 ms
spr_book_event.png	200	png	(index)	1.7 KB	47 ms
spr_book2.png	200	png	(index)	55.4 KB	44 ms
image.jpg	200	jpeg	(index)	176 KB	290 ms
image.jpg	200	jpeg	(index)	277 KB	294 ms
main.js	200	script	require.js:1962	9.2 KB	50 ms
jquery.min.js	200	script	require.js:1962	84.8 KB	94 ms
hbs.js	200	script	require.js:1962	22.1 KB	26 ms
component.min.js	200	script	require.js:1962	3.9 KB	33 ms
util.js	200	script	require.js:1962	978 B	23 ms
handlebars.js	200	script	require.js:1962	152 KB	160 ms
underscore.js	200	script	require.js:1962	36.2 KB	152 ms
json2.js	200	script	require.js:1962	12.0 KB	144 ms
flicking_component.js	200	script	require.js:1962	9.2 KB	56 ms

BEFORE

리소스 크기 : 500KB 이상

네트워크 요청 수 : 11회

시간 : 500ms 이상

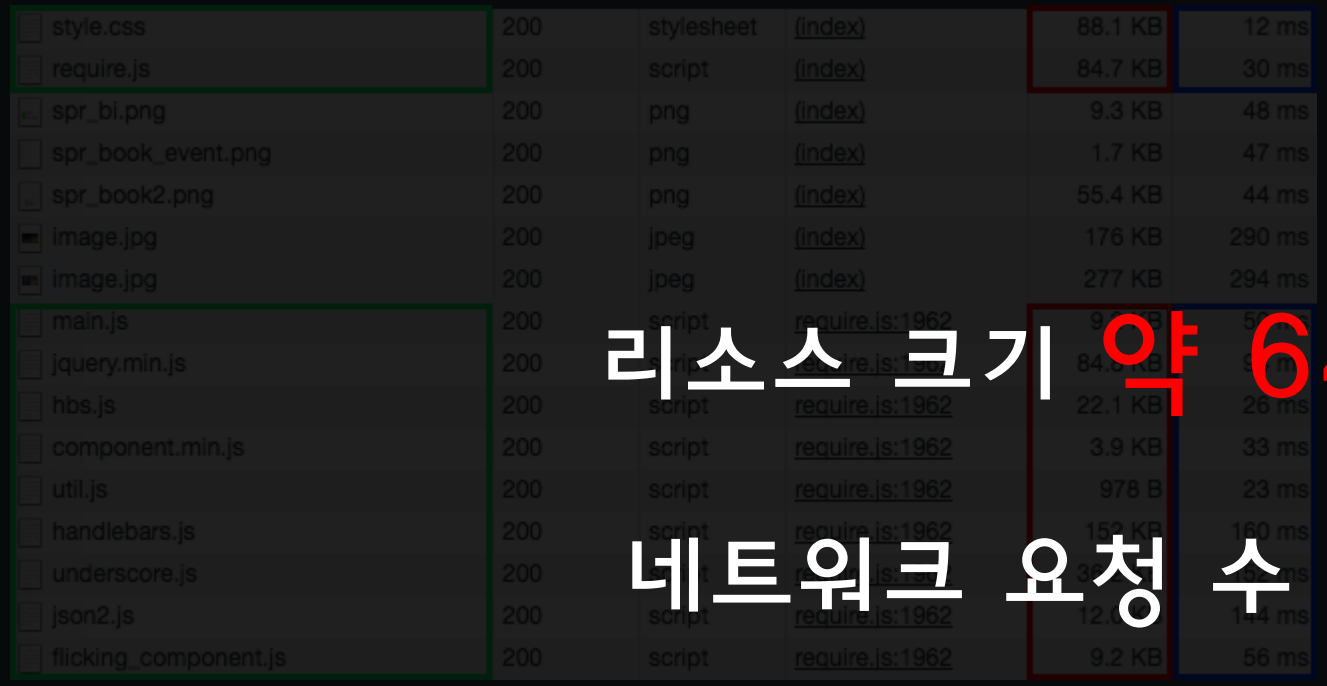
127.0.0.1	200	document	Other	5.7 KB	7 ms
style.css	200	stylesheet	(index)	71.7 KB	15 ms
main.bundle.js	200	script	(index)	108 KB	22 ms
image.jpg	200	jpeg	(index)	176 KB	228 ms
spr_bi.png	200	png	(index)	9.3 KB	14 ms
spr_book_event.png	200	png	(index)	1.7 KB	13 ms
spr_book2.png	200	png	(index)	55.4 KB	13 ms
image.jpg	200	jpeg	(index)	277 KB	224 ms

AFTER

리소스 크기 : 180KB

네트워크 요청 수 : 2회

시간 : 37ms 이상



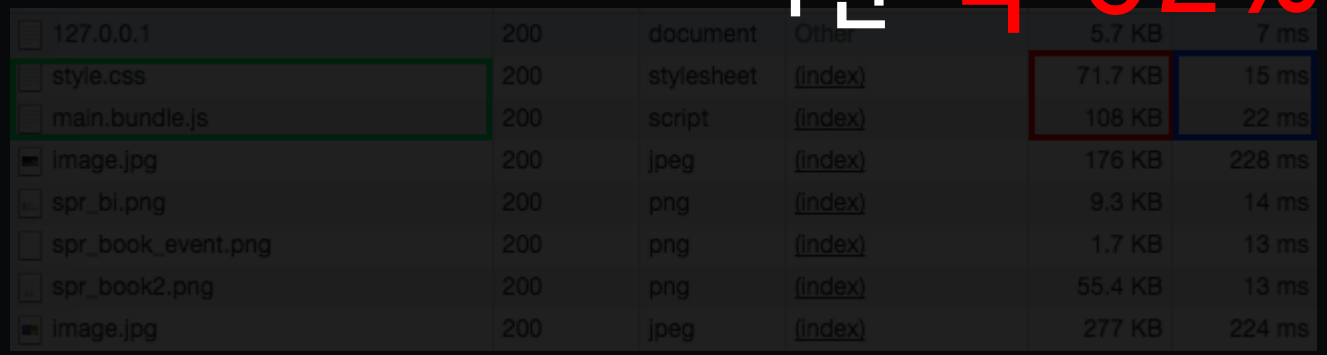
Before

리소스 크기 : 500KB 이상
시간 : 500ms 이상

리소스 크기 약 64% 감소

네트워크 요청 수 9회 감소

시간 약 92% 절약!



AFTER

리소스 크기 : 180KB
시간 : 37ms 이상

최종 성능 개선

Performance

These encapsulate your app's performance.

39

Metrics

These metrics encapsulate your app's performance across a number of dimensions

- First meaningful paint 5,610 ms
- First Interactive (beta) 6,410 ms
- Consistently Interactive (beta) 21,320 ms
- Perceptual Speed Index: 15,928 (target: < 1,250) 6
- Estimated Input Latency: 30 ms (target: < 50 ms) 100

Opportunities

These are opportunities to speed up your application by optimizing the following resources.

- Serve images as WebP 6,990 ms 1,283 KB
- Offscreen images 5,510 ms 1,011 KB
- Optimize images 4,210 ms 773 KB
- Properly size images 2,410 ms 442 KB
- Reduce render-blocking stylesheets 2,150 ms
- Reduce render-blocking scripts 1,950 ms
- Enable text compression 1,810 ms 332 KB

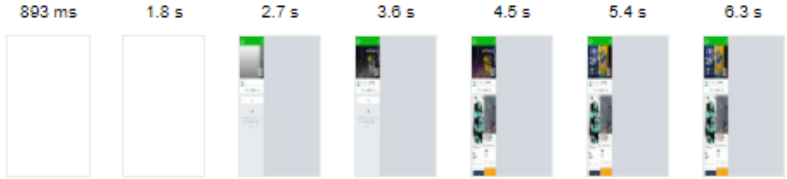
Performance

These encapsulate your app's performance.

92

Metrics

These metrics encapsulate your app's performance across a number of dimensions.



- First meaningful paint 2,030 ms
- First Interactive (beta) 2,790 ms
- Consistently Interactive (beta) 2,790 ms
- Perceptual Speed Index: 4,741 (target: < 1,250) 58
- Estimated Input Latency: 26 ms (target: < 50 ms) 100

Opportunities

These are opportunities to speed up your application by optimizing the following resources.

- Reduce render-blocking stylesheets 1,320 ms
- Enable text compression 740 ms 135 KB
- Serve images as WebP 520 ms 95 KB
- Optimize images 60 ms 10 KB

6 Passed Audits

- Reduce render-blocking scripts
- Properly size images
- Offscreen images

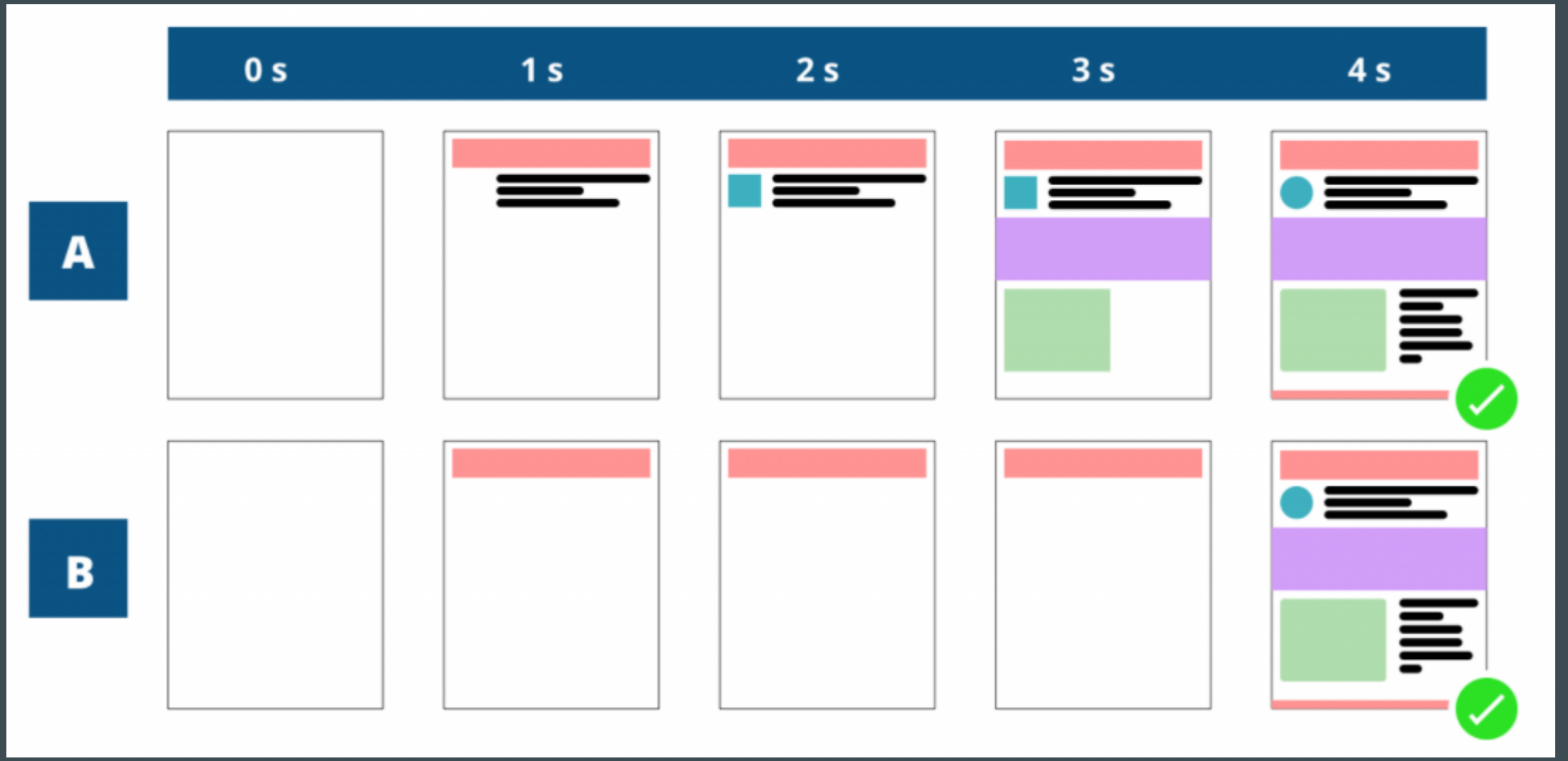
Speed Index

기존의 웹 성능 측정 방식
“브라우저가 onload이벤트에 도달하는 시간“

But, 이 수치는 사용자 경험이 반영되지 않는 방법이다.
그래서 나온 새로운 웹 성능 측정 방식 “Speed Index”

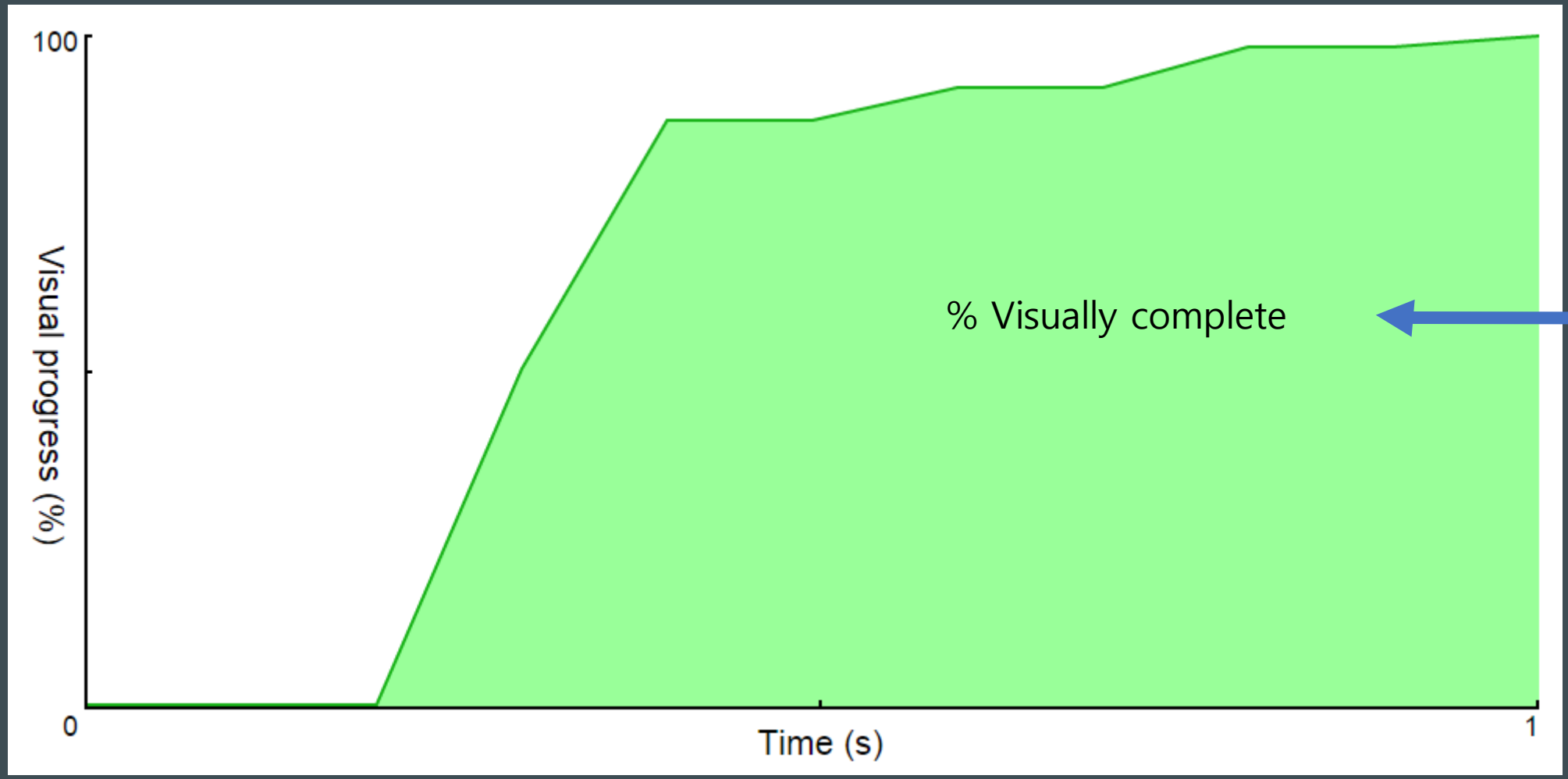
Speed index는
“실제 사용자의 경험을 포함한 평균 페이지 로딩시간”

Speed Index 예시

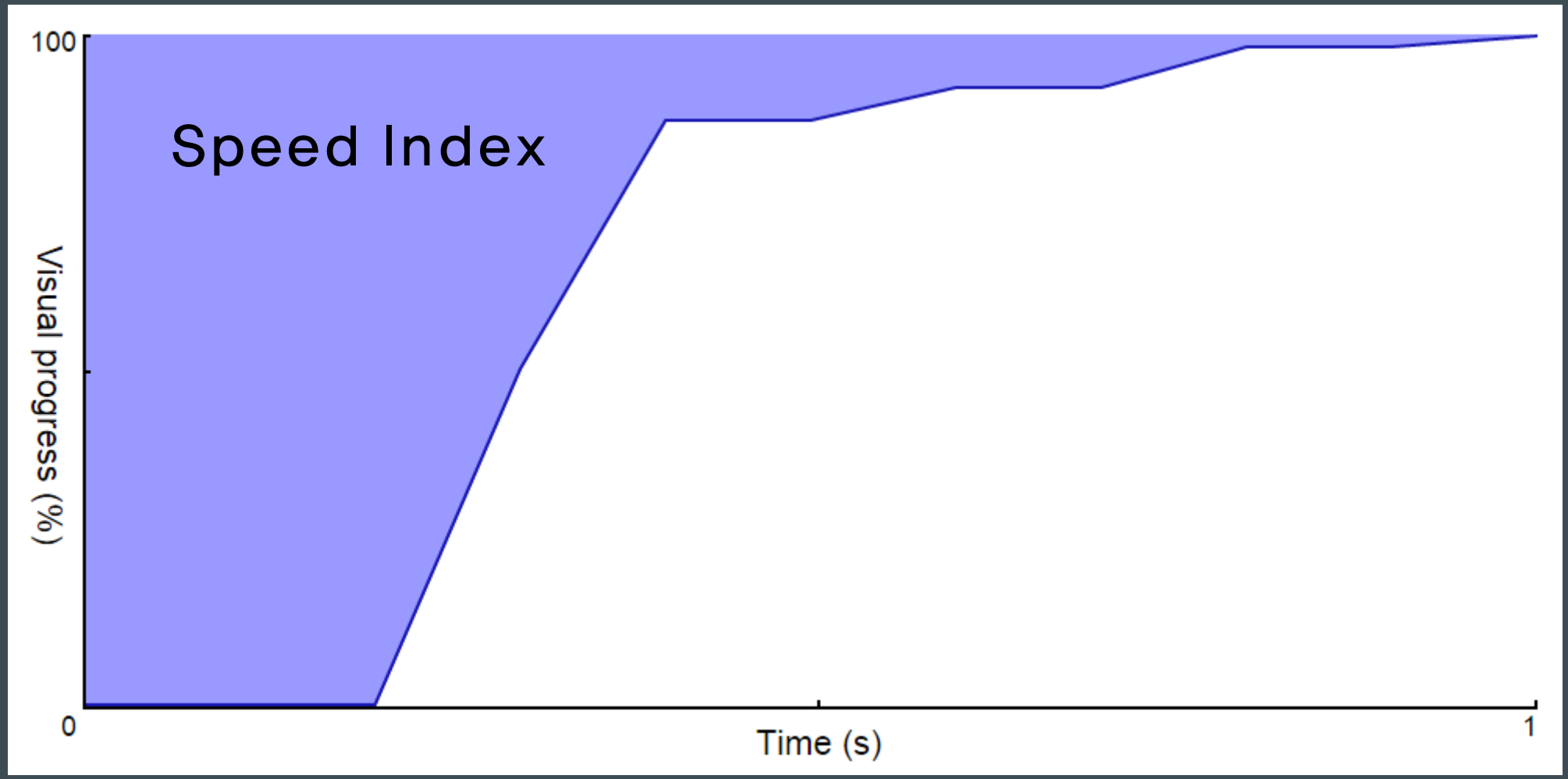


Speed Index 측정

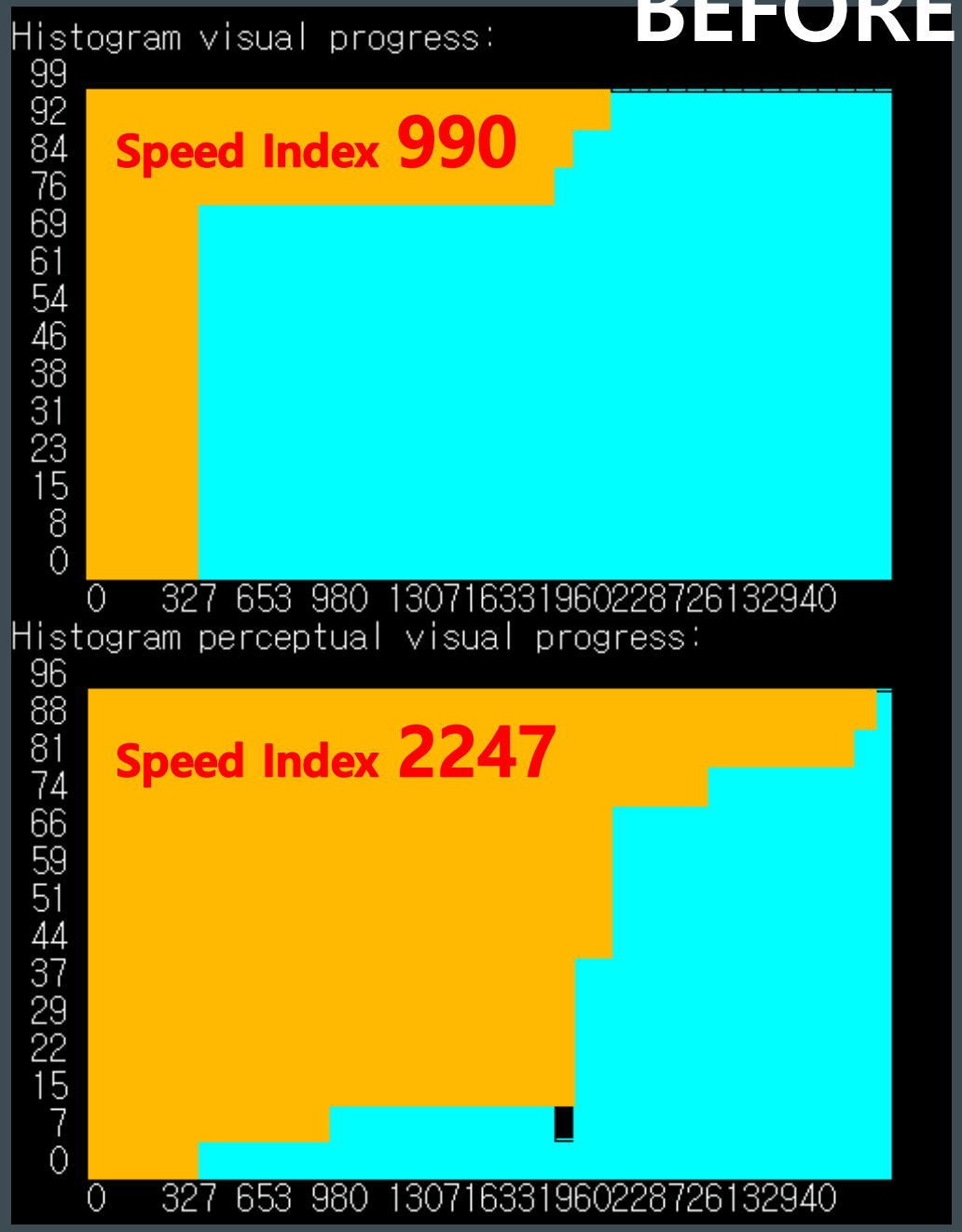
페이지 로드 될 때까지 시간 경과에 따라 시각적으로 완료되는 양



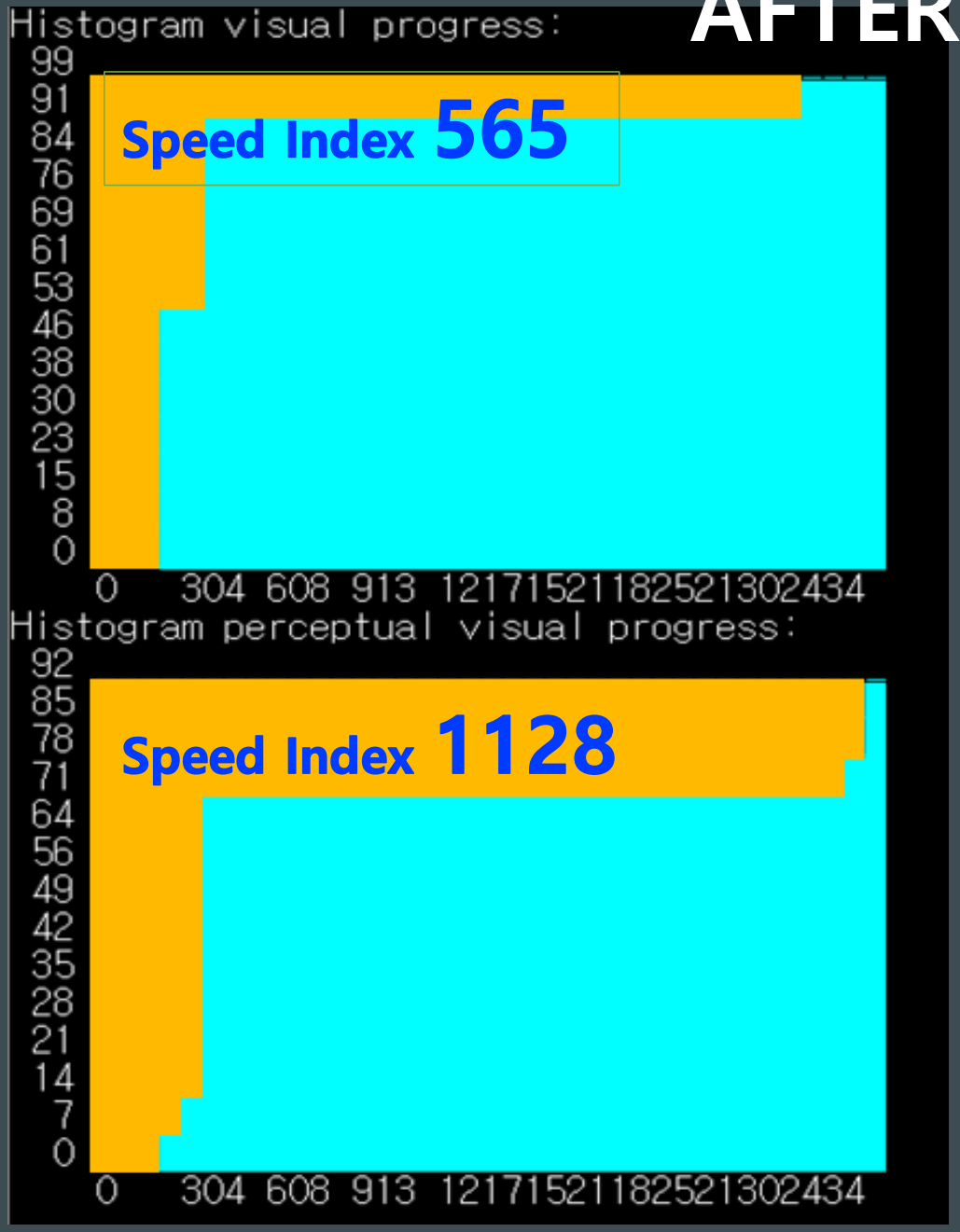
Speed Index 측정



BEFORE



AFTER



감사합니다.

Q&A

감사합니다.